The Behavior Analyst 1995, 18, 307–316 No. 2 (Fall)

Using Computers to Teach Behavior Analysis

Eliot Shimoff and A. Charles Catania
University of Maryland Baltimore County

When it is impractical to provide behavior analysis students with extensive laboratory experience using real organisms, computers can provide effective demonstrations, simulations, and experiments. Furthermore, such computer programs can establish contingency-shaped behavior even in lecture classes, which usually are limited to establishing rule-governed behavior. We describe the development of computerized shaping simulations and the development of software that teaches students to discriminate among reinforcement schedules on the basis of cumulative records.

Key words: computer software, simulations, shaping, reinforcement schedules, cumulative records

Designing a course in behavior analysis often involves compromises between behavior-analytic principles and practical considerations. For example, experimental analyses of the differences between contingency-shaped and rule-governed behavior suggest the importance of *doing* experiments (rather than simply reading about them), and courses in behavior analysis ideally should include laboratory experience (e.g., Karp, 1995). But when circumstances (e.g., large class enrollments, limited laboratory facilities) make such arrangements impractical, computer simulations and experiments can add important behavioral dimensions to the standard lecture format.

BEHAVIORAL CRITERIA FOR SOFTWARE EVALUATION

Designing and evaluating computer software should be an exercise in behavior analysis: Effective software reflects behavioral principles and incorporates appropriate contingencies to maintain behavior of both the student and the instructor. Evaluating software involves many criteria. Some decisions rely on skilled judgment: How realistic is a simulation of shaping? Sometimes the evaluation is based on whether the data are robust: Will every subject in an experiment on the serial learning of verbal material show a serial position curve? In other instances, evaluation requires empirical description and analysis: How long does it take for students to complete an activity, and is the level of difficulty appropriate?

The only way to develop the most effective software is through extensive experimentation: Start with a preliminary version, see where students get into trouble, then revise and retest. Software that has undergone extensive development is easy to recognize: Students have no trouble using it because the subtleties that were concerns for the programmer have been made invisible.

CONTINGENCIES FOR STUDENT BEHAVIOR

Effective software necessarily provides effective contingencies for the behavior of students: The programs are typically easy to use, instructions are clear and simple, and students learn what the program was designed to teach. Ease of use can only be assessed by studying behavior. Without experiments it is, for example, difficult to predict how many students will be confused by such apparently simple instructions as "use the arrow keys"

We dedicate this article with gratitude to Fred S. Keller, for everything he taught us about instruction.

Demonstration copies of *Behavior on a Disk* are available from CMS Software, P.O. Box 5777, Santa Fe, New Mexico 87502-5777.

Reprints may be obtained from either author at the Department of Psychology, University of Maryland Baltimore County, Baltimore, Maryland 21228-5398 (E-mail: shimoff@umbc.edu or catania@umbc.edu).

when there are as many as 14 arrows on the keyboard.

Does the program take into account the student's preexisting behavioral repertoires? Lecturers who misjudge their audience's skills and abilities can backtrack and elaborate; software must have such elaborations built in.

Is the software interactive? Interactivity simply refers to a relation between program flow and student response. Programs should present appropriate discriminative stimuli and require student responses to ensure that their behavior is under the control of the material being presented (i.e., does the program ensure that the student who completes the program has indeed mastered the material?). Sometimes this can be inherent in the program; students can complete a simulation of shaping only if they can shape the response. If a computer program involves an experiment with the data summarized in tabular format, the student can be asked questions about the table (e.g., number of trials needed for the subject to meet the criterion).

Additional constraints are imposed by the accessibility of appropriate computers (e.g., is multimedia hardware required?) and availability of the software (can the student use it at home, or must it be used in the campus computer center?).

CONTINGENCIES FOR INSTRUCTOR BEHAVIOR

Software cannot affect student behavior if it is not adopted by instructors. Developers of software must therefore minimize the response cost to instructors of adopting software and integrating it into a course. One way of minimizing some of these response costs is by having software distributed like other supplemental course material (e.g., workbooks, laboratory manuals).

Another set of contingencies involves integrating the students' completion of programs into the structure of a preexisting course. How does the instructor know that a student has completed a program? Even programs that are intrinsically interesting must compete with other activities, so assigning computer activities with no contingencies for completion is not likely to maintain much student behavior. Effective academic software should make it easy for the instructor to impose contingencies for completing the programs.

In what follows, we provide examples of computer programs on two topics in behavior analysis—shaping and reinforcement schedules—that meet the criteria outlined above (for reviews of software packages that include these programs, see Graf, this issue, Hyten, 1989, and Mulick, 1992).

SIMULATIONS OF SHAPING

Among the most useful sets of activities for students in behavior analysis are various simulations of shaping. Presumably, this reflects the importance of shaping as a skill with implications in a variety of behavioral settings, from the basic research laboratory to clinical interventions. We had been using a classroom demonstration of shaping (adapted from one originally created by B. F. Skinner; Catania, 1988, p. 476) that began with a rat pressing a lever with relatively low force. By gradually increasing the force requirement, we shaped more and more forceful presses, until the rat was pressing with its full body weight. The demonstration, which usually took about 15 min, was impressive, and our students often grunted in sympathy with and offered verbal encouragement to the rat. Still, they were primarily passive observers. Giving each of the hundred or more students a rat and access to the apparatus would have been ideal, but it was impractical. So we set out to develop a computer simulation.

The simulation was based on a distribution of possible response forces, with the mean of the distribution increasing when a response from the high end of the distribution was reinforced and decreasing when one from the low end was reinforced. The magnitude of the shift in either direction was proportional to the distance of the reinforced response from the mean of the distribution. In other words, reinforcing a response from the top end of the distribution produced the biggest increment in the distribution mean, and reinforcing one from the bottom end produced the biggest decrement. Responding began with a low distribution mean, and the student's objective was to win the game by shaping a response of at least 100 g (most students were unfamiliar with such technical units as Newtons). In our simulation, as in real life, limits were imposed by satiation (too many reinforcers and responding stops) and by extinction or strain (too many consecutive unreinforced responses or too small a proportion of reinforced responses and responding stops). If satiation or extinction intervened before successful shaping, the game was over. But win or lose, the student always had the opportunity to play again.

Several parameters must be set in such a simulation: the variability and initial mean of the response distribution, how much effect reinforcers have on the distribution, and the criteria for extinction and satiation. As outlined below, these parameters evolved on the basis of data collected from students.

The Shaping Game

The first of our shaping simulations presented a number representing the force of a response, and the student reinforced that response by typing R. The student's objective was to raise the response force from an initial low level of just a few grams to a terminal force of 100 g. (Note that this differs from the original classroom demonstration, in which the experimenter set a threshold force and then waited for a response. In this simulation, the response force was shown instead, and the experimenter had to decide whether or not to reinforce it.) The simulation ran

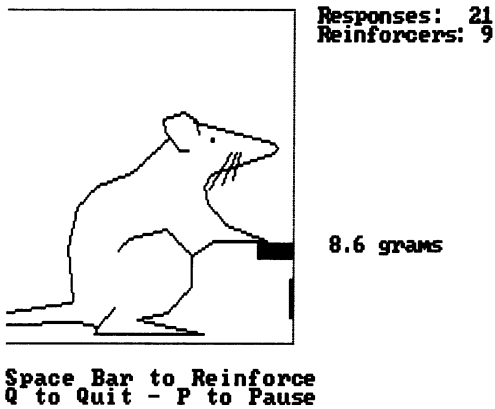

Figure 1. Sample screen from the Shaping Game. The rat has just pressed the lever with a force of 8.6 g. Simple graphics were used to enable the program to run on machines with only minimal graphics capabilities.

on the campus mainframe computer, and completion of the shaping simulation became a required part of our undergraduate course in learning. Dedicated real-time systems were not available then, so students could take their time in choosing whether or not to reinforce a given response, and when they sat at a terminal that printed out responses they could even examine prior sequences of reinforced and unreinforced responses. Only later, with the advent of dedicated personal computers (first the Apple II® and later MS-DOS® compatible machines), could the crucial realistic feature of having to react quickly to ongoing behavior in real time be incorporated into the simulation. At that point, program development began in earnest, and we added a modest graphics rat to the bare-bones presentation of response forces (see Figure 1).

One early refinement was to vary the starting point of the simulation so that students learned over several games that the absolute values of the first few responses did not predict the ease of shaping. Another refinement was converting response forces by a power function that varied from game to game, so that the effects of reinforcers on force were not linear over the course of a game (i.e., sometimes reinforcers produced large changes early in the game and small ones later, and sometimes vice versa). This reduced the likelihood that students would base their decision to deliver a reinforcer on specific numerical criteria (e.g., reinforcing any early response that exceeds 10 g). In addition, satiation and extinction criteria varied within a range from game to game, so students would not learn particular parameter values.

Over various revisions of the program, data were collected from students' plays of the Shaping Game. Evaluation included simple observations of students playing the game, discussions with players about the game, and various statistical analyses. One of the most useful analyses was a variation on signal detection analysis. Reinforcing a response in the upper half of the distribution was treated as a hit, whereas failing to reinforce it was treated as a miss; similarly, reinforcing a response in the lower half of the distribution was treated as a false alarm, whereas failing to reinforce it was treated as a correct rejection (Nevin, 1969). We found that proportions of hits and correct rejections by individual students rose over successive plays of the game. These measures allowed us to track the improvement with which students detected reinforceable responses over successive games. Other parameters and program details were revised and refined based on these and other quantitative data. For example, we took the number of games played until first success as the most appropriate measure of game difficulty, and the number of games played after students completed the plays required for the course as the most appropriate measure of whether the game was fun to play.

The game was eventually revised into four levels of difficulty: easy, medium, hard, and very hard. The levels varied not only in the magnitude of the effect of each reinforcer on the response distribution but also in the stringency of the satiation and extinction limits. Students typically began with the easy version and moved up to more difficult levels as their competence increased (a few students began and succeeded at the medium or hard levels). For this and for other shaping simulations, we aimed for programs that would establish successful shaping skills for each student within half an hour or less of computer time.

The shaping simulation had itself become an instrument of shaping: It shaped increasingly skilled shaping performances by our students. We have used the Shaping Game in both introductory courses and courses in learning, and most students win the easy version within three or four tries. They then become more sophisticated shapers by moving to the more difficult versions. However, the very hard version can be won only a small proportion of the time, even when played by very skilled students. Evidence that students find this version of the game especially challenging is that about 30% of the students in our introductory psychology classes go on to play and win the hard version even though they are only required to win the easy version.

The Threshold Game

Another simulation, the Threshold Game, was later developed to more closely match the contingencies in the original classroom demonstration. The player sets the minimum force that will produce a reinforcer; the program then generates a response from the distribution and reinforces that response only if it is above the threshold set by the player. The program tells the player whether the response was reinforced but does not display its force, so the player does not know whether the response just barely exceeded the threshold or exceeded it by a large amount. Similarly, in the case of an unreinforced response, the player does not know whether the response just missed the threshold or was far below it. In the Threshold Game, as in the Shaping Game, the player can lose the game either by raising the force threshold too much or too quickly, leading to strain and extinction, or by raising it too slowly, thereby leading to satiation while still at a low force level.

The Threshold Game has the feel of an actual demonstration, in which moving the force up so quickly as to produce pauses in lever pressing of just 2 or 3 min is tantamount to failure because it is difficult to maintain class interest when the rat is no longer pressing. In the simulation as in the actual demonstration, smaller increments in the threshold requirement are necessary as the target force is approached. Students playing the game seem to react to the simulated rat much as our earlier students reacted to a real rat, talking to it, making judgments about its intelligence (or lack thereof), and attributing to it many of the features of real organisms. (Interestingly, such attributions became even more accentuated when we later changed the Threshold Game to a simulation of a weight-lifting primate.)

But there is more to shaping than simply learning routines that may work only in particular cases. For that reason, we designed other simulations with different interactive properties, so that students would become sensitive to current response distributions and the effects of the reinforcers they delivered upon those distributions, rather than merely learning specific rules that might work only in specific instances (e.g., in the standard Shaping Game, students often learn limited and ultimately ineffective rules such as "Only reinforce a response if its force is higher than any one you have ever seen in the past").

Goal setting. One alternative version of the Threshold Game requires the player to increase the time a simulated student spends studying (from an initial studying bout of approximately 10 min to a goal of 2 hr) by setting increasingly higher goals. The underlying mathematical model is essentially the same as in the Threshold Game, but with an important difference. When shaping the force of a rat's bar press, the player must start with fairly large increases in the required force and then slowly decrease the step size; but when working with the simulated college student, the player must take fairly small steps in the beginning and require bigger steps towards the end. The student learns that successful shaping requires "knowing your organism," and that the goal setting so widely used in organizational settings is, in fact, an analogue to shaping procedures. Perhaps more significantly, the student's behavior is less likely to be superficially rule governed. We weighed the potential ambiguity of the reinforcing status of reaching a goal (goal setting probably always involves rule-governed behavior) against the importance of using a plausible shaping scenario that required a shaping strategy incompatible with the one established in the original Threshold Game.

Shaping and Response Distributions: The Distribution Game

In the original classroom demonstration with a real rat, as well as in the Threshold Game and goal-setting simulations described above, the experimenter sets the threshold and delivery of the reinforcer is automatic whenever the next response exceeds the threshold. In most instances of shaping, however, the contingencies are much more like the original Shaping Game, in which the organism responds and the experimenter must decide whether or not to reinforce that response.

At this point, we decided that it would be useful for students to know a little about the theoretical basis of the shaping model we had used. So we developed a simulation in which the student shapes the location at which a mouse pokes its nose up at the bottom of the computer screen. The mouse starts at the far left, and the shaper's task is to move it over to the far right. This version differed from the original Shaping Game in that reinforcers acted on local response probabilities rather than on the distribution mean.

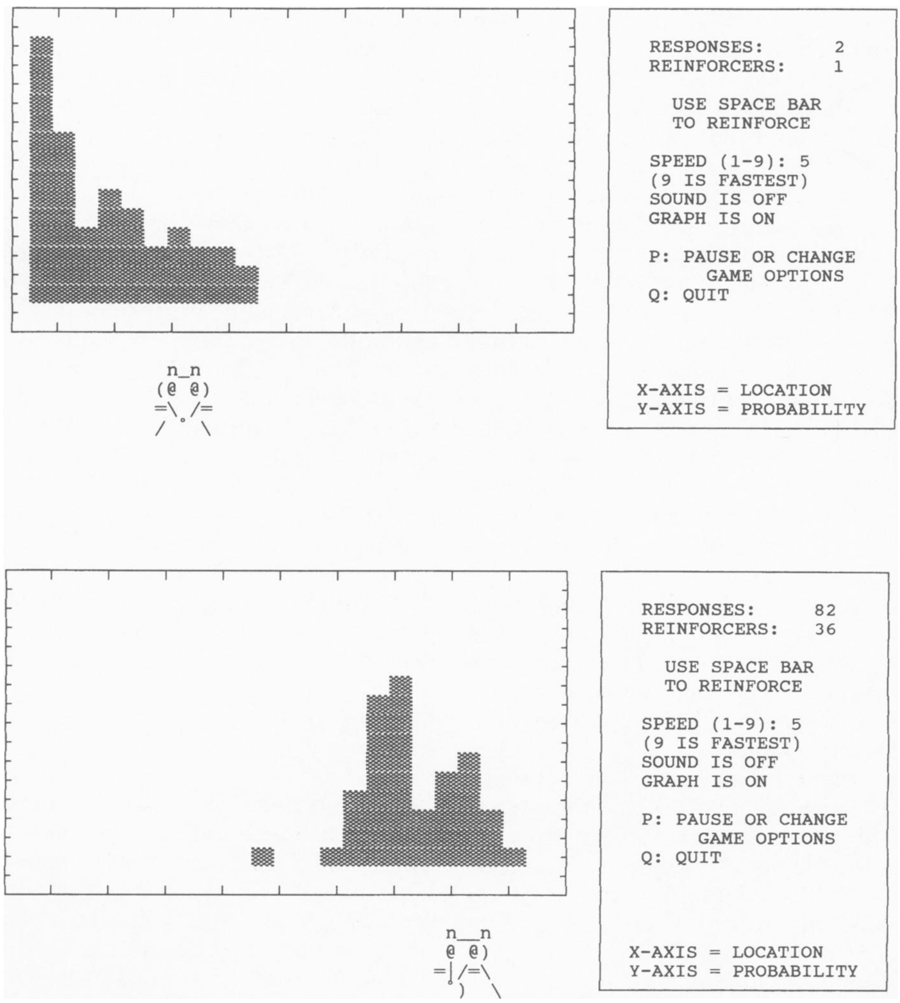

Figure 2. Sample early and late screens from the Distribution Game simulation of shaping, which includes probability distributions of the locations at which a mouse pokes its nose. In the top screen, the distribution has been shifted only slightly from its starting range on the left, and the mouse has just popped up near the right end of the distribution. In the bottom screen, the distribution has been moved most of the way across the screen by differential reinforcement (after 82 responses and 36 reinforcers), and the mouse has just popped up near the right end of the distribution. Just a few more judiciously delivered reinforcers are needed to win this game.

The main innovation was that the response distribution was made visible to the player, in the form of bars above the places where the mouse can appear, with heights proportional to the probability that the mouse will appear there. If the shaper reinforces a response, probability increases at and around that location; if a response occurs without being reinforced, probability at and around that location decreases. (It did not seem possible to design an adequate simulation of shaping without including the effects of nonreinforcement as well as reinforcement in the region of the distribution at which the response occurred.) Sample screens from one version of this simulation are shown in Figure 2.

In the Distribution Game, as in the earlier simulations, satiation and extinction are built into the program. Unreinforced responses and reinforcers that occur late (thereby reinforcing "other behavior") both reduce response probability. Extinction occurs when response probability has gone to zero at all possible locations. In addition, a reinforcer delivered too late reduces relative probabilities across the entire distribution, consistent with a reinforcer that follows behavior other than the nose poke (i.e., behavior that occurs elsewhere, and is therefore incompatible with nose poking).

Students who win this simulation are well prepared for discussions of shaping as an example of the selection of behavior. They also learn that repeated reinforcers at locations where response probability is already high do not move the distribution; instead, reinforcers have the biggest effects on responses of low probability at the extremes of the distribution. In this game, in fact, shaping is most efficient (in the sense of producing the biggest changes with the fewest reinforcers) when response probabilities remain generally low, so that outlying and relatively low-probability responses become relatively more likely. It remains to be seen how consistent this simulation is with actual instances of shaping (e.g., Eckerman, Hienz, Stern, & Kowlowitz, 1980; Galbicka, Kautz, & Jagers, 1993).

Verbal shaping. While we were developing these shaping simulations, our laboratory research on the shaping of college students' verbal behavior (e.g., Matthews, Catania, & Shimoff, 1985) suggested another variant of the Shaping Game. In this simulation, students learn to shape positive self-references by a depressed client who frequently makes extremely pessimistic statements (e.g., "The world is infinitely awful" or "Everything seems loathsome") but occasionally utters somewhat less pessimistic statements (e.g., "Things are really pretty bad" or "I feel rotten"). The player's job is to judiciously reinforce increasingly more positive statements to get the client to exclaim eventually that "Everything is perfect." The player reinforces a statement by pressing the space bar, and the program then generates an accepting therapist comment, such as "Uh-huh" or "Indeed!" or "I see." The underlying algorithm for this simulation is essentially the same as that for the Distribution Game. But judging the positiveness of verbalizations is more difficult than judging response force, so it was necessary to relax the contingencies on satiation and extinction (in either case, when the game is lost the client leaves for another therapist).

The program links groups of utterances scaled to different levels of pessimism or optimism to different locations on the distribution, and selects the successive utterances presented to the player in proportion to response probabilities in that distribution. The program also occasionally presents irrelevant utterances (e.g., "My canary is very thoughtless"). If irrelevant utterances are reinforced, the distribution of response probabilities is unaffected but the reinforcer counts toward the satiation criterion. This shaping simulation extends the concept of a response continuum to a verbal dimension, and the student who masters it is well prepared for a discussion of Truax's (1966) analysis of Rogerian therapy as verbal shaping.

Relevance

It is difficult to collect data on the extent to which experience with shaping simulations prepares students for the shaping of the behavior of a real organism more effectively than lectures or texts. Even if the simulations are shown to be superior, the outcome might depend only on inadequacies of the lectures or the text materials. Nevertheless, on the basis of the performances of undergraduates who have moved into our laboratory after completing these simulations, we are willing to assert that students who have shaped the force of a simulated rat's lever press, the study time of a simulated student, the weight lifting of a simulated ape, the nose pokes of a simulated rat, and the verbalizations of a simulated depressed client know far more about shaping than those who have merely watched us shape the force of a rat's bar press in a class demonstration and read and listened to us lecture about the theoretical and practical aspects of shaping.

TEACHING ABOUT REINFORCEMENT SCHEDULES

Teaching about reinforcement schedules is often difficult; there is a lot of material, and many subtle discriminations must be learned. Because the cumulative record is a common tool for illustrating the effects of reinforcement schedules on behavior, we decided our first task was to teach how a cumulative recorder works. Lecturing on the topic usually took a long time, and students' responses to quiz questions revealed that many failed to understand the relation between the slope of the record and response rate.

Here again, we found that a carefully designed computer program could be very helpful. The evolution of these programs was guided in part by an experimental analysis of verbal behavior under the discriminative control of schedule contingencies (Catania, Shimoff, & Matthews, 1989). The first part of our program demonstrates the relation between slope and response rate by presenting a short segment of a cumulative record on the computer screen and requiring students to superimpose records of their own responding on that sample by pressing the space bar. The deviation of the student's match to the sample record determines whether the student has to try again or can move on to a new sample. The eight sample records in this portion of the program advance from relatively steady responding at low, moderate, and high rates to various patterns, including acceleration, deceleration, break-and-run responding, and scallops.

Every student must master this portion of the program before moving on

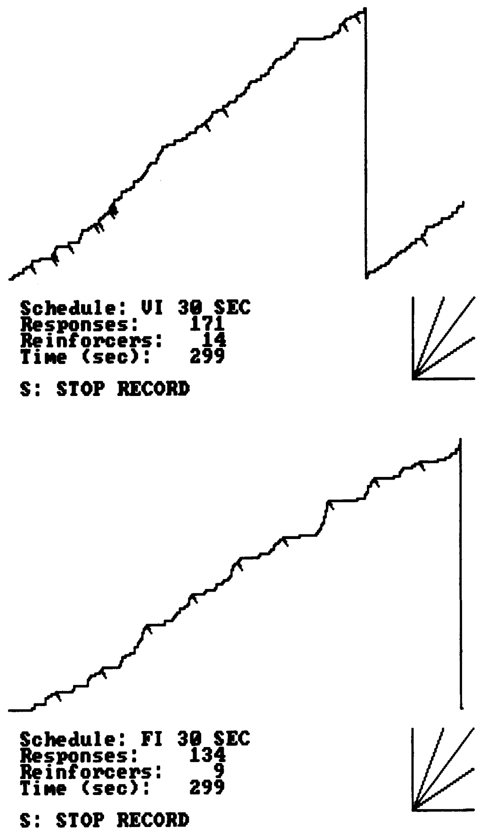

Figure 3. Sample screens of simulated variable-interval (top) and fixed-interval (bottom) performances from the cumulative-record program.

to the next part, which is designed to teach the student to discriminate among the cumulative records characteristically generated by four basic schedules: fixed ratio, fixed interval, variable ratio, and variable interval. The student selects a schedule and its parameter value and watches a cumulative record generated by the computer. (It is not necessary to let a student sit for a full minute to see a scallop emerge within a single 1-min fixed interval. We learned early that we could reduce complaints substantially by generating the records at a speeded-up rate rather than in real time.) Sample fixed-interval and variable-interval screens are shown in Figure 3. It is worth noting that, unlike the idealized records that appear in typical undergraduate texts, these records resemble those in Ferster and Skinner (1957).

After seeing some sample records, the student requests the record generated by an unknown schedule and selects the relevant schedule name from a multiple-choice list. The program that generates these records was designed to produce variability and to display characteristic effects of schedule parameters (e.g., successive fixed-interval scallops vary in curvature, fixed-ratio postreinforcement pauses are a function of ratio size). Students complete this portion of the program by meeting a criterion of nine correct identifications for the 10 most recent records.

The third and final component of the cumulative-record program teaches another discrimination that is crucial to a student's understanding of schedules. If the student's own responses (presses on the space bar) produce consequences (pips on the student's own record), can the student identify the schedule contingencies by which those pips are being delivered? Students can either select particular schedule contingencies to see how they work or select an unknown schedule and then attempt to identify it from a multiple-choice list. Early examples include only the four basic schedules of the earlier component of the program, but other contingencies are added as students get more proficient (including fixed-time and variable-time schedules, differential-reinforcement-of-low-rate schedules, and avoidance). Our students are often particularly impressed by what they learn about how to distinguish variable-ratio from variable-interval contingencies, and their discrimination between fixed-interval and differential-reinforcement-of-low-rate contingencies is particularly helpful in preparing them for lectures and discussions on these topics.

Students who complete this program learn how cumulative records work and can discriminate among different schedule performances as well as among different schedule contingencies. Although students do not become verbally sophisticated about schedules solely on the basis of using these programs, they are more likely to answer examination questions about schedules accurately, and they understand schedules in a very different way than students whose experience with schedules is purely intraverbal (e.g., those who have merely learned statements such as "fixed-interval schedules maintain scallops").

CONCLUSION

The first and most obvious audiences for software that teaches behavior analysis are students in courses in learning and behavior analysis. But behavior analysis need not, indeed should not, be restricted to courses in behavior analysis. Academic software provides a vehicle for subtly introducing principles of behavior analysis into the introductory psychology course. One way instructors can do this is to have introductory psychology students complete some activities similar to those provided to students in behavior analysis courses. For example, we routinely have introductory psychology students play simple versions of the Shaping Game.

But a more pervasive aspect of behavior analysis permeates almost any successful software package. As noted earlier, effective software must be easy to use without demanding substantial class time. That more or less precludes group experiments in which instructors must collect data from many students and present statistical analyses. Thus, successful software will probably involve experiments on individual organisms (typically with the student as the subject), one of the hallmarks of experimental design in behavior analysis.

Researchers in behavior analysis have become sensitive to the difference between rule-governed and contingency-shaped behavior (e.g., Shimoff, Matthews, & Catania, 1986), and there is much to be said for ensuring, as much as possible, that behavior-analytic skills be contingency shaped. Computers can be important and effective tools for establishing sophisticated behavior that appears to be contingency shaped even in classes that do not provide real (unsimulated) laboratory experience.

REFERENCES

Catania, A. C. (1988). Problems of selection and phylogeny, terms and methods of behaviorism. In A. C. Catania & S. Harnad (Eds.), *The selection of behavior: The operant behaviorism of B. F. Skinner* (pp. 472–488). New York: Cambridge University Press.

Catania, A. C., Shimoff, E., & Matthews, B. A. (1989). An experimental analysis of rule-governed behavior. In S. C. Hayes (Ed.), *Rule-governed behavior: Cognition, contingencies and instructional control* (pp. 119–150). New York: Plenum.

Eckerman, D. A., Hienz, R. D., Stern, S., & Kowlowitz, V. (1980). Shaping the location of a pigeon's peck: Effect of rate and size of shaping steps. *Journal of the Experimental Analysis of Behavior, 33,* 299–310.

Ferster, C. B., & Skinner, B. F. (1957). *Schedules of reinforcement.* New York: Appleton-Century-Crofts.

Galbicka, G., Kautz, M. A., & Jagers, T. (1993). Response acquisition under targeted percentile schedules: A continuing quandary for molar models of operant behavior. *Journal of the Experimental Analysis of Behavior, 60,* 171–184.

Hyten, C. (1989). A review of CMS Software's *Psychology On A Disk. The Behavior Analyst, 12,* 227–232.

Karp, H. J. (1995). Rat lab for fun and profit. *The Behavior Analyst, 18,* 147–154.

Matthews, B. A., Catania, A. C., & Shimoff, E. (1985). Effects of uninstructed verbal behavior on nonverbal responding: Contingency descriptions versus performance descriptions. *Journal of the Experimental Analysis of Behavior, 43,* 155–164.

Mulick, J. A. (1992). Review of *Behavior On A Disk* from CMS Academic Software: Instructional programs for teaching teachers. *Research in Developmental Disabilities, 13,* 301–307.

Nevin, J. A. (1969). Signal detection theory and operant behavior: A review of Green and Swets' *Signal Detection Theory and Psychophysics. Journal of the Experimental Analysis of Behavior, 12,* 475–480.

Shimoff, E., Matthews, B. A., & Catania, A. C. (1986). Human operant performance: Sensitivity and pseudosensitivity to contingencies. *Journal of the Experimental Analysis of Behavior, 46,* 149–157.

Truax, C. B. (1966). Reinforcement and nonreinforcement in Rogerian therapy. *Journal of Abnormal Psychology, 71,* 1–9.